TURBO news

Revista para Computadores ATARI® Nº 5 - Noviembre 1989

$ 550

CONOCIENDO TU JOYSTICK
MODIFICANDO EL SET DE CARACTERES
CURSOS

ATARI

ADQUIERELOS EN LOS SIGUIENTES PUNTOS DE VENTAS

• **ANTOFAGASTA:** COOPERCARAB KW VIDEO LA ESPANOLA • **VIÑA DEL MAR:** FALABELLA VINA INSIS MPR COMPUTACION • **VALPARAISO:** COMPUTRONIC • **SANTIAGO:** AUDIO BICICLETA INTERNAC CASA ROYAL CENTRO ATARI COMERCIAL ESTADO COMPUMANQUE COMPUCENTER FALABELLA AHUMADA FALABELLA P. ARAUCO IMACO INFOGROUP PC STORE PETERSEN ROLEC SUPERMERCADOS UNIMARC TASCO VIDEO CLUB INTERNACIONAL • **RANCAGUA:** CASA ZUNIGA • **CURICO:** MULTIHOGAR • **TALCA:** LIBRERIA "EL AHORRO" MULTICENTRO VIDEO CLUB CASSAL • **CHILLAN:** CASA EDISON • **CONCEPCION:** COOPERCARAB DISMAR DISMAR 2 EQUS PHANTER RAPSODIA SESCO • **LOS ANGELES:** DISTRIBUIDORA MERINO • **ANGOL:** SCORPIO • **VICTORIA:** CASA SIGMUND • **TEMUCO:** COMERCIAL MANQUEHUE ESTABLECIMIENTOS GEJMAN FALABELLA • **PUCON:** ELTIT • **VILLARRICA:** JOYERIA KETTERER • **VALDIVIA:** ELECTROMUSICA • **LA UNION:** IMPORTADORA COSMOS • **OSORNO:** CASA REAL FOTO EXPRESS • **PUERTO VARAS:** ELECTRO HORN • **PUERTO MONTT:** COMERCIAL MANQUEHUE DIMARSA • **COYHAIQUE:** FACI HOGAR • **PUNTA ARENAS:** BALFER LTDA.

CONTENIDO

Entrevista Atari en la Escuela	2
Turbo Mail	4
Basic (Lección 5)	6
Atari por Dentro Set de Caracteres	12
Desprotector y Protector de Listado	15
Assembler (Lección 5)	16
Ranking del mes Descripción de Juegos	20
Logo (Lecci ón 3)	22
Desarrollando Hardward Trucos con el Joystick	25
Atari Sounds	29

Circulación mensual. Nacional e Internacional
Destinada a los usuarios de computadores ATARI (R) como material didáctico de PROGRAMACION TURBO news (R) es una publicación de EDITORA TURBO LTDA. Domicilio Av. Holanda 2456 - Teléfono: 2238063
SANTIAGO CHILE.

EDITOR: Editora Turbo Limitada. **PRODUCCION**: Pedro P. Caraball Alvarez - Marcelo A. Waldbaum Olszevicki - Mauro Pieressa Schachtel, Programadores y Diseñadores de Computación. **REPORTAJE:** Silvia Edelstein. **COLABORACION:** Pedro P. Caraball Alvarez. **ARTE/DISEÑO:** Odalí Guerrero López. **FOTOGRAFIAS:** Isabel Bellalta. **CORRECCION:** Marcial Valenzuela S. **COMPOSICION:** Oriber Ltda.

DEPARTAMENTO DE VENTAS Y PUBLICIDAD: Hemán Vittini G. Agradecemos la colaboración de: Cocha S.A. Centro Atari (Augusto Leguía Sur 75). Prof. Emilio Antileff. **ATARI**, es marca registrada de Atari Corporation. **TURBO**, es marca registrada de **EDITORA TURBO LIMITADA** (Reg. Marc. Nº342428 9-05-89). Impresa en los talleres de Impresos Nova Ltda. quien actúa como Impresora.

EDITORIAL

Queridos amigos, iniciamos aquí un nuevo contacto donde nuevamente pondremos todo nuestro empeño para transmitirles nuestros conocimientos adquiridos luego de tantos años de ejercitarnos con este fabuloso amigo: el computador ATARI. Y cada vez lo hacemos con más fuerza ya que recibimos día a día el estímulo de tus cartas y colaboraciones. Notamos en ellas que Uds. poseen el mismo cariño e inquietudes que nosotros renovamos todos los días y, por ello, nos sentimos gustosos de ayudarlos y orgullosos de que se hayan volcado en nosotros, para satisfacerlas.

Continuamos en este número con los cursos dirigidos a los tres niveles de programadores: Logo para niños, Basic para principiantes y Assembler para los más expertos. Tampoco descuidamos el desarrollando Hardware ni el conocimiento del Atari por dentro. El Ranking del mes continúa recibiendo tus votos y preparándonos para el Gran Ranking del año, que publicaremos en el número de enero, en el cual estás invitado a participar.

Esperamos que una vez más esta edición cubra todas tus expectativas. Que te diviertas!!!

ESTUDIANTES Y PROFESORES AT

ATARI EN L

por: SILVIA EDELSTEIN

TURBO NEWS estuvo de visita en varias escuelas en las que funcionan Talleres de Computación ATARI. Queríamos analizar la importancia de la computación en la formación intelectual de los niños recabando la opinión de maestros y educandos. Para ello, conversamos con el Prof. Emilio Antileff, docente a cargo de esta experiencia comunal.

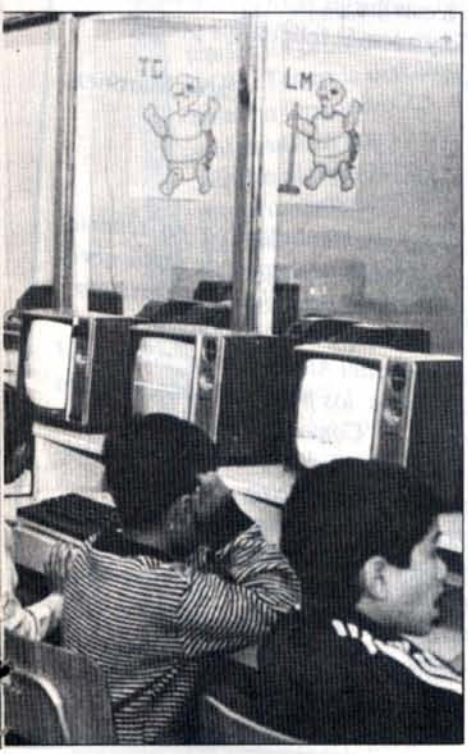
El mismo nos cuenta que el Proyecto de Computación de Peñalolén comenzó en 1988, a partir de fondos de la Subsecretaría de Desarrollo del Ministerio del Interior. El objetivo era financiar el equipamiento de un laboratorio con 20 computadores cada uno, para las Escuelas Básicas 211 y 187 de esa comuna. El proyecto abarcó también una escuela media, en la cual se amplía la formación de los alumnos con el uso de programas utilitarios para facilitar su salida laboral.

La Escuela que visitamos en primer lugar, la 171, financió su Taller con recursos propios. Decidieron equiparse con computadores ATARI en virtud de la calidad de su material educativo y del apoyo que proporciona Coelsa en materia docente y asuntos técnicos.

Los talleres de computación para los alumnos se organizaron

RAIDOS POR LA COMPUTACION

ESCUELA



en grupos pequeños que van rotando. Los niños de la comuna provienen de familias trabajadoras, en las que reciben menor estimulación y oportunidades con respecto a niños de clases más aventajadas.

Los docentes refieren que estos niños han incrementado notablemente algunos aspectos de su proceso de aprendizaje desde que acceden al uso de computadores en su formación. Se ha verificado un cambio importante en su capacidad de atención y concentración. El ausentismo y la mala conducta en clase disminuyeron. Se muestran reacios a abandonar el taller de computación, donde permanecen horas absortos en la pantalla de sus equipos.

Los niños modificaron también sus patrones de interacción. Se muestran más comunicativos entre sí y se consultan y aconsejan permanentemente durante las clases de computación, compartiendo los logros y avances en el aprendizaje.

La enseñanza en el Taller se basa en el lenguaje LOGO. Coelsa entrega el asesoramiento técnico al docente, mediante la intervención del psicopedagogo Pablo Saba Cadenas y su equipo de colaboradores. El docente, luego, elabora las guías de enseñanza que el alumno puede llevar a su hogar.

Luego de escuchar el preciso informe del Prof. Antileff, quisimos ver con nuestros propios ojos el funcionamiento de los talleres. Cuando entramos al

primero de ellos, el de la Escuela 171, nos sorprendimos al ver a los niños tan concentrados y entretenidos en su tarea, aun en ausencia del docente, que conversó con nosotros durante largo rato. Organizamos una rueda de charla para conocer sus opiniones acerca de esta nueva experiencia, todos los niños tienen bien claro la existencia de computadores en todas las áreas del trabajo humano. Mencionan oficinas, bancos, hospitales, laboratorios y hogares. Ante la pregunta acerca de su futura profesión, casi todos desean realizar algún tipo de tareas vinculadas con la computación. Un gran número de los niños desearía trabajar como secretarias, empleados de Bancos u oficinas o dedicarse a la enseñanza. De computación, por supuesto.

ATARI EN LA ESCUELA

Observando sus trabajos en la pantalla, se advierte un rico despliegue de imaginación: barcos, helicópteros, casas y ciudades. Los trabajos son individuales pero todos desean asesorar o consultar a sus compañeros con el objeto de mejorar el resultado final colectivo. En el taller reina la paz y la cordialidad, el compañerismo y los buenos modales. Ni siquiera la visita del periodista y su fotógrafo, separan la atención del monitor.

Junto con el Prof. Antileff, visitamos luego otras dos escuelas básicas, la 211 y la 187. Los Talleres de Computación ATARI en ambas estaban en pleno funcionamiento. Los niños graficaban en sus pantallas y los docentes satisfacían dudas y problemas. Y quien los observa piensa que realmente ha comenzado una nueva era. El computador que asombra y atemoriza a los adultos es ya un "viejo amigo" para los niños.

Si sabemos que en poco tiempo ninguna clase de trabajo podrá llevarse a cabo sin la ayuda de la informática, qué mejor que familiarizar a los pequeños con el computador.

En las escuelas que visitamos hoy, hay un proyecto positivo en marcha.

TURBO MAIL

Por: Marcelo Waldbaum

CONEXION AL MONITOR:

Tengo un equipo Atari 800 XL y un monitor de video Computexto, al cual quiero conectarlo pero no he encontrado la forma de hacerlo. En casa de un amigo vi su revista TURBO news y me decidí a escribirles para preguntar cómo hacerlo y además felicitarlos por su edición.

Pablo Martínez

Pablo:

Seguramente tu amigo no tendría nuestra Revista del mes de septiembre, pues en ella desarrollamos un artículo en el cual explicamos todas las conexiones del Atari. En su página 24 describimos los pines del conector para el monitor. Como podrás ver, el pin Nro. 4 es la salida del video compuesto. Con este pin y el Nro. 2 que es la tierra puedes conectarlo con la ficha correcta a tu monitor en la ficha de entrada de video y si tu monitor tiene además sonido, con los pines 2 y 3 puedes obtener la salida del audio.

MANTENTE EN CONTACTO

PROBLEMAS CON LOS D.O.S.:

Antes de ir al motivo de mi carta deseo felicitarlos por el reciente lanzamiento de su revista TURBO news. La encuentro muy educativa para nosotros los usuarios de computadores ATARI.

En primer lugar, me gustaría plantearles un problema que tengo con mi DOS. Tengo la disketera Atari 1050 y el Sistema Operativo 2.5, mientras que un amigo utiliza con su Drive el DOS 3. ¿Existe alguna forma de compatibilizar mis programas con los suyos? En segundo lugar, me gustaría que me expliquen cómo debo alinear el lápiz de luz con el software de graficación.

Miguel Guerrero.

Miguel:

Gracias por tus felicitaciones.

Para poder compartir programas con tu amigo, puedes utilizar un programa utilitario que seguramente encontrarás en el disco que viene con tu disquetera. Se llama COPY32.COM y convierte archivos del DOS 2.5 al DOS 3 y viceversa. Pero lo mejor sería que convenzas a tu amigo de utilizar cotidianamente el DOS 2.5 pues, a nuestro criterio, es más completo.

Con respecto a la alineación del TURBO LIGHT-PEN, muchos usuarios tienen el mismo problema. Cuando en la pantalla te aparece una barra vertical, debes posicionar el lápiz sobre ella y no tratar de llevar el cursor hasta la línea, pues en este caso estarías alineando incorrectamente el lápiz de luz.

COMPATIBILIDAD CON EL 1040ST:

Estoy planeando reemplazar mi antiguo Atari 600-XL por un 1040ST y me gustaría saber si puedo seguir utilizando mi impresora Epson, así como también todo el software que tengo desarrollado en el XL.

Sergio Torres.

Sergio:

El Atari 1040 ST está basado en el procesador Motorola 68000 el cual no tiene ninguna similitud con el 6502 que utiliza tu 600 XL. Por lo tanto, tendrás que traducir el software que tienes ya desarrollado para que pueda ejecutarse en tu ST. Lo que sí podrías compartir, son los archivos de textos, los cuales puedes transmitir de un computador al otro utilizando dos Modems para realizar la transferencia. En cuanto a tu impresora Epson, podrás utilizarla tranquilamente siempre y cuando sea una impresora paralela. La mayoría de las Epson lo son, pero existen modelos como la HomeWriter 10 que está especialmente diseñada para los Atari XL. Si la tuya es paralela, lo único que necesitarás es el cable adecuado.

BASIC

LECCION 5

En este número veremos uno de los conceptos más importantes en programación: el uso de Vectores, una nueva forma de variables que nos facilitan la estructura de datos en nuestros programas.

Por: Mauro Pieressa S.

Hasta ahora, habíamos hablado de variables numéricas y alfanuméricas. Recordemos que éstas eran "cajas" vacías a las que les asignábamos un valor cualquiera. Pero qué pasa, por ejemplo, si en un programa de agenda telefónica, deseamos ingresar los nombres de todos los compañeros de nuestra escuela. Deberíamos usar una variable para cada uno de ellos. Supongamos, además, que en realidad la agenda es de todas nuestras amistades y su número es indeterminado porque todos los días conseguimos nuevos amigos. Cada vez habría que incorporar más variables!! Pero eso no es todo. Desde el punto de vista de la programación, tendríamos que escribir cada vez, la instrucción INPUT, PRINT o cualquier operación que deseemos hacer con dichas variables, haciendo la programación muy engorrosa y los programas super-largos. Para evitar todo lo anteriormente mencionado, es que se han creado los Vectores. Estos son una sucesión de variables, todas con el mismo nombre, individualizadas por un subíndice o número identificatorio encerrado entre paréntesis.

Los Vectores, al igual que las variables, pueden ser numéricas o alfanuméricas, diferenciándose por terminar estas últimas con el signo $. Los Vectores numéricos se utilizan de la siguiente manera:

Con la instrucción DIM, la misma que servía para crear las variables alfanuméricas, le indicamos al computador el nombre del Vector y la cantidad máxima de valores que él mismo puede contener. Ejemplo:

```
DIM TELEFONOS (50)
```

Esto significa que podremos almacenar hasta 50 números telefónicos completos (cada uno de ellos con todos los dígitos que hagan falta), en dicho Vector. Pero no 50 números individuales como sería el caso si éste hubiese sido alfanumérico. Si queremos imprimir un número telefónico en particular, ejecutamos la siguiente instrucción:

```
PRINT TELEFONOS (2)
```

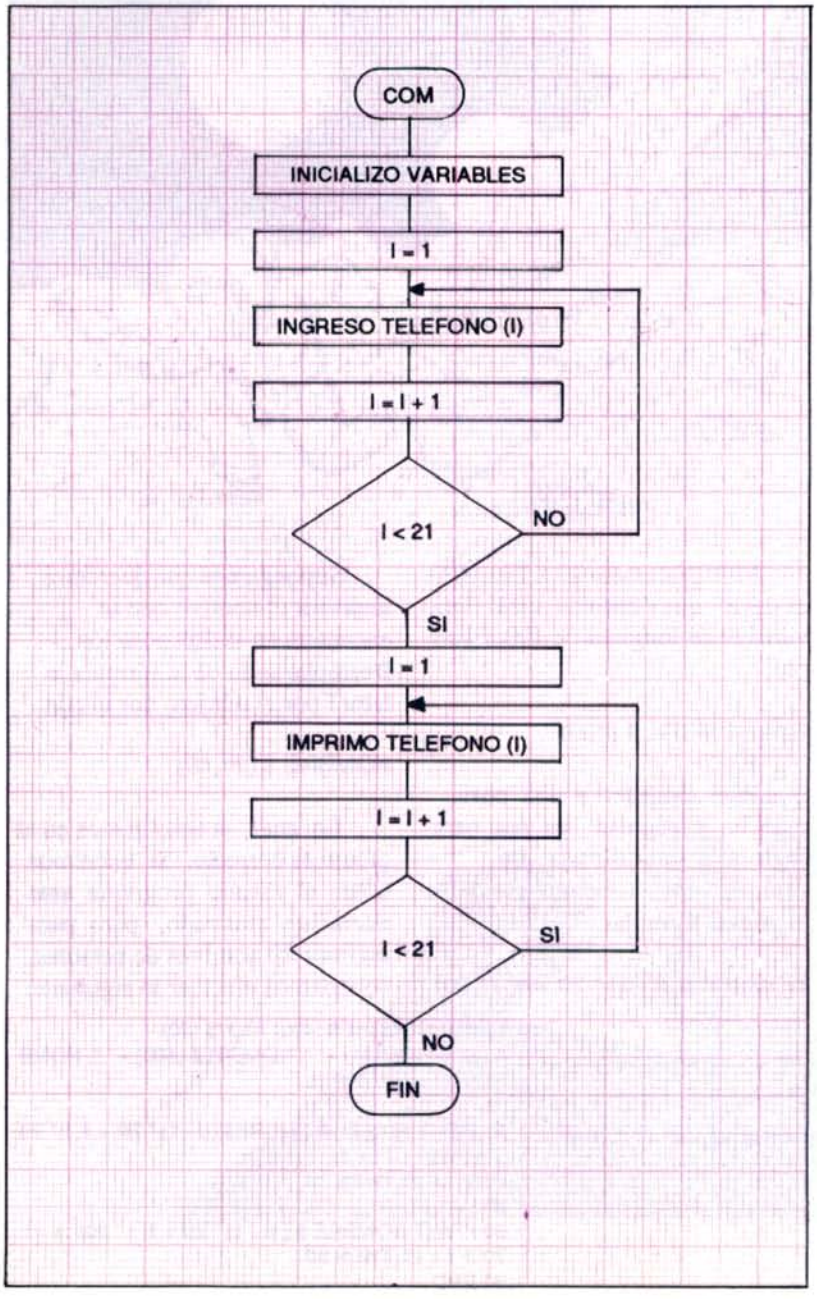
Entonces va a imprimir el número telefónico que se encuentra en segunda posición. Para ingresar números con la instrucción INPUT se puede hacer lo mismo. Ejemplo:

```
INPUT AUX: TELEFONOS (2) = AUX
```

En la posición dos, se va a almacenar el número que ingresemos.

La gran ventaja de los Vectores consiste que el subíndice puede, a su vez, ser una variable. Con ello se logra que con una sola instrucción PRINT o INPUT se pueda imprimir o llenar cualquier Vector completamente. Esto se ve en el siguiente ejemplo:

```
10 DIM TELEFONOS (20)
20 I = 1
30 INPUT AUX: TELEFONOS (I) = AUX
: I = I + 1
40 IF   I < 21   THEN 30
50 I = 1
60 PRINT TELEFONOS (I) : I = I + 1
70 IF I < 21  THEN 60
80 END
```



Al ejecutar este programa con la instrucción RUN, se nos pedirá que ingresemos 20 números telefónicos y posteriormente los imprimirá. Para ingresar menos de 20 números en forma variable (1,4 ó 15), hay que cambiar la instrucción que controla el fin de los ingresos, es decir, la línea número 40. Hay que poner una condición de fin. Por ejemplo: que ingresen números hasta encontrar un número telefónico negativo. La condición de cierre debe ser una que sepamos que en la realidad no se va a dar nunca, como es un número telefónico negativo o cero.

El programa quedaría de la siguiente manera:

```
10 DIM TELEFONOS (20)
20 I = 1
30 INPUT AUX: TELEFONOS (I) = AUX
(I) : I = I + 1
40 IF TELEFONOS (I - 1) > 0 AND I < 21
THEN 30
50 J = 1
60 PRINT TELEFONOS (J) : J = J + 1
70 IF J < I THEN 60
80 END
```

Estudiemos las modificaciones. En la línea 40 preguntamos si el número telefónico que se encuentra en la posición I - 1 es negativo o si llegamos al final, es decir, la posición 21. Debemos preguntar por la posición I - 1 y el número 21, porque la instrucción antes de la pregunta a la que se hace referencia, la 30 incrementa a I en 1 o sea que la posición que habíamos ingresado y la que tiene el último valor es la I - 1.

En el caso de los Vectores alfanuméricos el tema es un poco más complicado. Vimos que cuando hacíamos:

DIM TELEFONOS (20)

Estábamos definiendo 20 variables que pueden contener números telefónicos con cualquier cantidad de dígitos cada uno de ellos. En cambio si hacíamos:

DIM TELEFONO $ (20)

(Variable alfanumérica) estábamos definiendo UNA variable que puede poseer hasta 20 números de un dígito. Para convertir esta variable en un Vector nos valdremos del siguiente artificio. Supongamos que vamos a almacenar 20 nombres de amigos y que la longitud de dichos nombres tiene un máximo de 20 letras. Esto se ve claramente en el siguiente dibujo.

Multiplicamos 20 x 20 = 400, es decir, que debemos crear una variable de longitud de 400. Ejemplo:

DIM NOMBRE $ (400)

Para acceder a una posición en particular, ya sea para ingresarla, modificarla, imprimirla, etc., hacemos uso de la siguiente fórmula:

NOMBRE $((I - 1) * 20 + 1, I * 20)

Supongamos que queremos imprimir el nombre que se encuentra en la 2da.posición. Reemplazando en la fórmula la letra I por el número dos queda:

NOMBRE $ (21,40)

Lo cual, si nos fijamos en el dibujo,es correcto. Si queremos hacer el mismo programa anteriormente mostrado, pero para ingresar 20 nombres de personas, deberíamos efectuar lo siguiente:

| | | |

Nótese que lo único que nemos modificado son las instrucciones 30 y 60 y el resto se ha mantenido igual. Para borrar un Vector alfanumérico y comenzar a trabajar, por ejemplo, podemos utilizar la misma técnica enseñada para borrar una variable alfanumérica. Recordemos que se hacía de la siguiente manera:

NOMBRE $ (1) =" ": NOMBRE $ (400) = " ": NOMBRE $ (2) = NOMBRE $

Sin embargo para borrar un Vector numérico debemos poner en cero (0) cada una de sus posiciones. Por ejemplo:

```
10 I = 1
20 TELEFONO $ (I) = 0
30 I = I + 1
40 IF I < 21 THEN 20
50 END
```

Para concluir vamos a explicar dos limitaciones que tienen los Vectores. La primera de ellas es que no se puede poner como variable de una instrucción INPUT, una posición determinada del Vector. Ejemplo:

INPUT NUMERO (2)

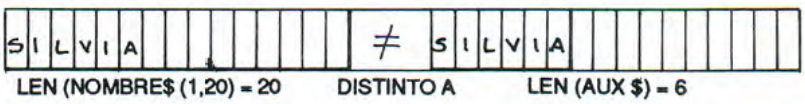
Para solucionar esto nos valemos de una variable auxiliar y luego,mediante una asignación, le damos el valor correspondiente a la posición del Vector. Ejemplo:

INPUT AUX: NUMERO (2) = AUX.

El segundo problema es que cuando utilizamos un Vector alfanumérico, cada una de sus posiciones tiene como longitud (al aplicársele la instrucción LEN) la máxima que se le ha asignado. Por ejemplo, en el caso del Vector NOMBRE $ anteriormente mostrado, cada una de sus posiciones tiene longitud 20. Sin embargo, habíamos visto que al asignar a una variable alfanumérica cualquiera, un valor, mediante la instrucción INPUT, la misma tenía una longitud igual a la cantidad de letras o números ingresados. Por ejemplo, si definimos una variable AUX $ de longitud 20 y mediante una instrucción INPUT le asignamos una palabra de longitud 6 (Silvia por ejemplo), el LEN nos va a dar justamente 6. El problema que surge se ve claramente en el siguiente ejemplo: Supongamos que queremos buscar en un Vector llamado NOMBRE $, el nombre Silvia, que se encuentra en la posición 1, por ejemplo, e ingresamos en una variable AUX$ el nombre que queremos buscar o sea Silvia. Si comparamos NOMBRE$ (1,20) con AUX $, a pesar de tener ambas variables el mismo contenido el resultado va a indicar que son distintos. La solución consiste en hacer que la variable AUX$ tenga la misma longitud que NOMBRE$ (1,20) para lo que debemos agregarle espacios en blanco a la derecha para hacer que no varíe su contenido. Ejemplo:

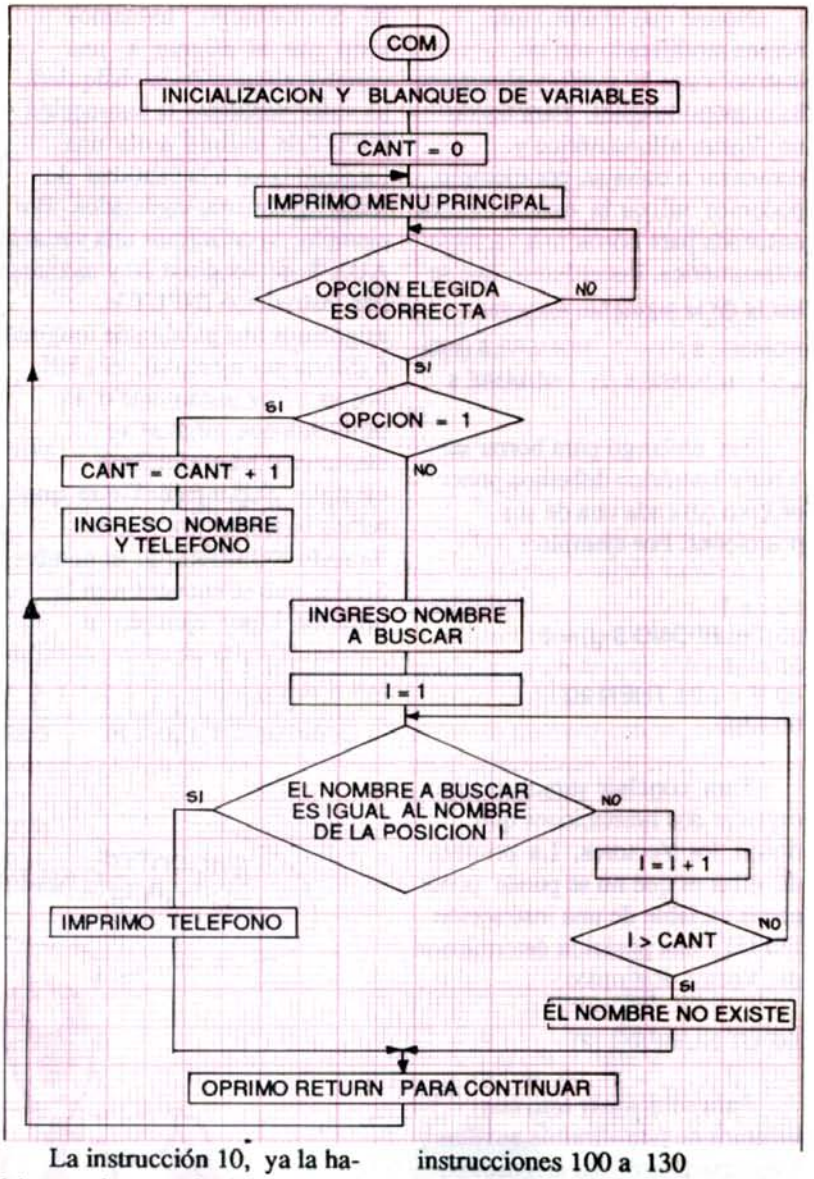
AUX$ (LEN (AUX$) + 1,20) = "

Ponemos 20 espacios en blanco, que es el valor máximo posible, así nos aseguramos de



que no vaya a faltar ninguno y los que sobran simplemente no se tienen en cuenta.

Para concluir, veremos cómo se hace una agenda telefónica. Estudiaremos primero el diagrama de flujo.



La instrucción 10, ya la habíamos visto en el número anterior y la utilizamos para definir las variables. En la 20 blanqueamos el Vector NOMBRE$ y ponemos en cero la variable CANT, que va a representar la cantidad de nombres que posee nuestra agenda telefónica. Vimos que el vector numérico teléfono, no hacía falta blanquearse de la instrucción 30 a la 90, imprimimos el Menú Principal y solicitamos que se ingrese la opción elegida, verificando que su contenido sea correcto, es decir 1 ó 2. En las instrucciones 100 a 130 ingresamos los nombres y teléfonos correspondientes. Como hay una cantidad CANT de nombres ya ingresados, le sumamos 1 a esa variable para ingresar el nombre actual a la posición correspondiente.

En las instrucciones 300 a 400, realizamos la búsqueda de cualquier nombre solicitado. Primero, en la instrucción 300 ingresamos el nombre a buscar y realizamos el truco explicado anteriormente para que la variable

NOM$ tenga la misma longitud que cualquiera de las componentes del Vector NOMBRE$. En la línea 310, inicializamos a I en 1. Esta es la variable que vamos a utilizar para recorrer todo el Vector hasta llegar a la posición que corresponda. En la 320, comparamos a NOM$ con el nombre que se encuentra en la posición I. Si coinciden los nombres, entonces imprimimos el número telefónico correspondiente. Si no coincide incrementamos a I para ver si es igual al siguiente. No debemos descartar la alternativa de que el nombre no exista. Esto lo hacemos comparando que la variable I no haya excedido la cantidad de nombres que tenemos almacenados, lo que hacemos en la línea 330. Si lo excede, significa que el nombre no existe pues ya hemos recorrido todo el Vector sin encontrarlo. Finalmente, oprimimos la tecla RETURN para continuar. La misma es utilizada para detener momentáneamente el programa y darnos tiempo para ver el resultado de la búsqueda. Notarás que cada vez que detienes el programa y lo vuelves a hacer funcionar, todos los nombres hasta ese momento ingresados habrán desaparecido. Para evitar esto, existen los medios de almacenamiento (cassetera o diskettera) cuya utilización explicaremos en números siguientes. HASTA PRONTO!!!

PROGRAMA

```
1 REM ******************************
2 REM *   REVISTA TURBO NEWS (R)   *
3 REM *       MAURO PIERESSA       *
4 REM *       CURSO DE BASIC       *
5 REM *                            *
6 REM * OBJETIVO: Programa de ejem- *
7 REM * plo de aplicacion de vecto- *
8 REM * res.                       *
9 REM ******************************
10 DIM NOMBRE$(800),TELEFONO(40),NOM$(
20)
20 CANT=0:NOMBRE$(1)=" ":NOMBRE$(800)=
" ":NOMBRE$(2)=NOMBRE$
30 PRINT "{
          MENU PRINCIPAL"
40 PRINT "↓↓↓↓↓1)CARGAR TELEFONOS"
50 PRINT "↓↓2)CONSULTAR TELEFONO↓↓↓↓"
60 TRAP 60:PRINT "ELIJA OPCION";:INPUT
 OPCION
70 IF OPCION<1 OR OPCION>2 THEN 60
80 IF OPCION=1 THEN 100
90 GOTO 300
100 CANT=CANT+1
110 PRINT "{INGRESE NOMBRE: ";:INPUT N
OM$:NOMBRE$((CANT-1)*20+1,CANT*20)=NOM
$
120 TRAP 120:PRINT "↓INGRESE TELEFONO:
 ";:INPUT FONO:TELEFONO(CANT)=FONO
130 GOTO 30
300 PRINT "{INGRESE NOMBRE A BUSCAR: "
;:INPUT NOM$:NOM$(LEN(NOM$)+1)="
                                "
310 I=1
320 IF NOMBRE$((I-1)*20+1,I*20)=NOM$ T
HEN PRINT "↓↓EL NUMERO ES: ";TELEFONO(
I):GOTO 400
330 I=I+1:IF I>CANT THEN PRINT "EL NOM
BRE NO EXISTE":GOTO 400
350 GOTO 320
400 PRINT "↓↓OPRIMA RETURN PARA CONTIN
UAR";:INPUT NOM$:GOTO 30
```


CompuCenter

- ATARI
- COMMODORE
- APPLE

ATENCION TODOS LOS DIAS DEL AÑO

PARQUE ARAUCO
LOCAL 247 - A
FONO: 2420596

- Equipos
- Suministros
- Software
- Materiales didácticos
- Programas IBM-MACINTOSH

SET DE CARACTERES

ATARI POR DENTRO

Por: Marcelo Waldbaum

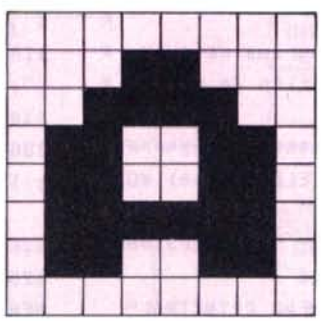
Cuando encendemos nuestro computador, vemos en la pantalla la palabra READY.
Para que el computador pueda desplegar mensajes de textos, existe en el Sistema Operativo una definición de caracteres llamados Set.

Al imprimir el computador la palabra READY, en realidad, lo que está haciendo el Sistema Operativo, es almacenar en la zona de memoria que representa la pantalla, los Bytes 82 69 65 68 89. Estos números en realidad, representan a las letras R E A D Y y es el Antic el encargado de desplegarlos.

Para hacerlo, el Atari posee en su ROM una tabla de correspondencia entre Bytes y caracteres. Cada carácter en el conjunto está constituido por puntos en una matriz que tiene ocho puntos de ancho y ocho de altura. Cada uno de estos 64 puntos puede activarse o no, con lo cual definimos la textura de un caracter.

Esta definición de caracteres se convierte en 8 Bytes que lo definen por filas desde arriba hacia abajo. Si queremos definir la letra A tendremos la siguiente configuración de Bits:

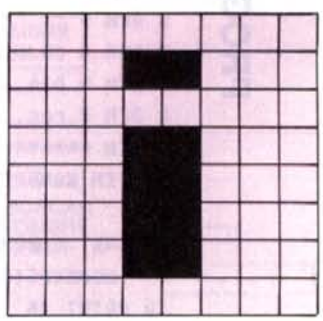
DEFINICION DE LA A



BINARIO	DECIMAL
00000000	0
00011000	24
00111100	60
01100110	102
01100110	102
01111110	126
01100110	102
00000000	0

Siguiendo esta estructura, podemos modificar los caracteres para generar nuevos o bien para dibujar en la pantalla. Como podrás ver, el computador no tiene en su teclado los dos símbolos de admiración "¡ y !", sino que sólo posee el "!". Es posible que en un programa que estemos desarrollando nos interese que en la pantalla salga el símbolo que no está definido, para lo cual debemos generarlo de la siguiente forma:

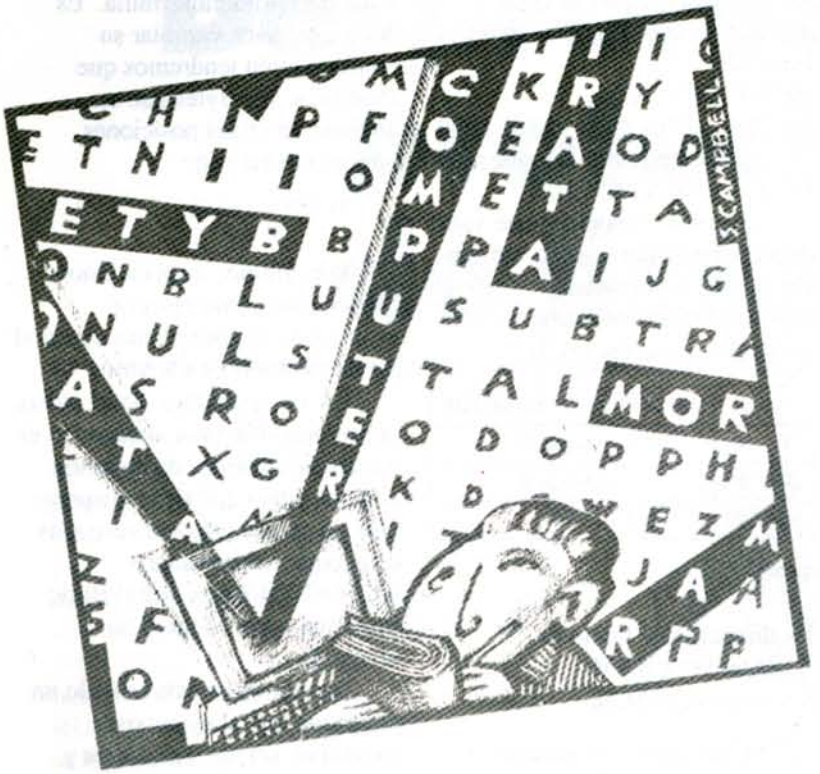
DEFINICION DEL ¡



BINARIO	DECIMAL
00000000	0
00011000	24
00000000	0
00011000	24
00011000	24
00011000	24
00011000	24
00000000	0

La información en cada definición de caracter se almacena en la tabla del ROM como una serie de 8 Bytes, representando cada uno de ellos, una fila horizontal de 8 puntos. Puesto que hay 8 Bits en un Byte, si un Bit está en 1, determina que el punto correspondiente esté activado en la pantalla y si su Bit está en 0, en la pantalla aparecerá desactivado.

Existen 128 caracteres distintos en el conjunto de caracteres Atari. Es decir que tomando 8



Bytes cada uno para su definición, necesitamos 1024 Bytes para el set completo. Debido a que cada caracter puede desplegarse en la pantalla en video inverso, existen 128 x 2 = 256 caracteres posibles de visualizar. Pero el computador no requiere de una tabla especial para los videos inversos, pues los obtiene automáticamente, seteando en 1 todos los Bits 0 del caracter y viceversa.

Esta tabla de 1024 bytes se encuentra en el ROM dentro de las posiciones de memoria comprendidas entre $E000 y $E3FF o en decimal entre 57344 y 58367. Si analizamos el contenido de los 8 primeros Bytes de la tabla, veremos que son todos 0, pues en la primera posición del set de caracteres se encuentra definido el espacio en blanco cuya configuración lógica es todos los puntos desactivados.

El sistema de caracteres desarrollado en el ATARI, es uno de los más potentes, pues permite la redefinición de los mismos por el usuario. Además en su ROM, nuestro computador posee un set de caracteres opcional que es el Set Internacional de Caracteres. En él podemos encontrar los caracteres utilizados en los distintos idiomas, las vocales con acentos y también la letra "ñ".

Es la intención de este artículo que podamos utilizar en nuestros desarrollos, tanto el set de caracteres internacionales como así también, la definición de nuestros propios caracteres.

Muy relacionada con los caracteres se encuentra la posición de memoria 756 en decimal. En ella le definimos al computador de dónde debe sacar la tabla de definición de caracteres para desplegarlos en la pantalla.

Cuando el computador se enciende, esta posición contiene el valor 224 ó bien en hexadecimal $EO. Esto le está indicando que su tabla de caracteres se encuentra definida en $E000. Es decir, que siempre los sets de caracteres deben comenzar en una dirección múltiplo de 256, pues si multiplicamos 224 * 256 obtendremos 57344 ó $E000.

En memoria podemos tener cuantos sets de caracteres podamos almacenar en RAM y con la posición de memoria 756 podemos ir cambiando el set utilizado. En las posiciones de memoria comprendidas entre $CC00 y $CFFF se encuentra ubicado el Set de Caracteres Internacionales. Es decir, que si "pokeamos" el valor $CC ó 204 en la posición de memoria 756, estaremos autorizando al Sistema Operativo a cambiar el set de caracteres.

Como ejemplo, enciende tu computador y presiona las teclas CONTROL y B simultáneamente. Luego ingresa la instrucción:

```
POKE 756,204
```

Verás que el caracter "I" se convierte en una ñ. Esto se produjo por el cambio del set de caracteres, del standard al internacional.

Como ejemplo de la utilización de los sets de caracteres, vamos a cambiar el caracter que se obtiene al presionar CONTROL y la tecla B por una

SET DE CARACTERES

ñ. Para esto, primero debemos copiar todo el set de caracteres del ROM en una zona reservada de la RAM, modificar la definición del CONTROL B y luego, cambiar el contenido de la posición 756, para que apunte al nuevo set.

Para ubicar rápidamente la dirección de los 8 Bytes que debemos modificar para redefinir un caracter, primero debemos saber su código de ATASCII. Para esto, podemos consultar la tabla de caracteres que se incluye en el Manual de Instrucciones del computador o bien utilizar la instrucción BASIC ASC.

Por ejemplo, para saber el código ASCII de la letra A, deberás ingresar en el Basic el siguiente comando:
PRINT ASC("A")

y el computador te contestará 65, que es su ATASCII.

Una vez conocido este valor deberás consultar alguna de las siguientes 3 fórmulas dependiendo en qué rango se encuentre el valor ASCII.

ASCII	UBICACION EN MEMORIA
0 - 31	I + (A+64) *8
32 - 95	I + (A-32) *8
96 - 127	I + (A*8)

donde:

I = dirección de inicio de la tabla en RAM.
A = valor ATASCII.

Si deseamos modificar el caracter Control B, cuyo valor en ATASCII es 2, obtendremos que sus 8 Bytes de definición están en:

I=(2+64) *8, pues debemos utilizar la primera fórmula. Es decir que para cambiar su configuración tendremos que modificar los Bytes que se encuentran en las posiciones comprendidas entre:

I+528 y I+535.

Por último, explicaremos brevemente cómo reservar espacio de memoria para ubicar el set de caracteres en RAM.

La computadora Atari utiliza la posición 106 para almacenar el número de páginas de memoria que considera que tiene disponibles. Si a este valor le restamos 4, estamos reservando 4*256=1024 Bytes que el Basic reservará y por lo tanto no afectará.

Incluimos en este artículo un programa que al ejecutarse cambia el set de caracteres y modifica el Control B por una ñ y te invitamos a enviarnos tus nuevos sets de caracteres para que podamos publicarlos.

PROGRAMA

```
100 REM RESERVO 1024 BYTES PARA
110 REM ALMACENAR EL SET
120 REM
130 MEMTOP=PEEK (106) -4
140 POKE 106, MEMTOP
150 GRAPHICS O
160 REM
170 REM CHROM = $E000 SET STANDARD
180 REM CHRAM SET EN RAM
190 REM
200 CHROM = PEEK (756) *256
210 CHRAM = MEMTOP *256
220 REM
230 REM LISTO EL PROGRAMA PARA QUE SE
240 REM VEA COMO SE VA DEFINIENDO EL
250 REM SET EN RAM.
260 REM
270 LIST 320, 540
280 REM
290 REM DEFINO AL SISTEMA OPERATIVO
300 REM QUE EL NUEVO SET ESTA EN
310 REM MEMTOP
320 REM
330 POKE 756, MEMTOP
340 REM
350 REM COPIO EL SET DE ROM A RAM
360 REM
370 FOR I = 0 TO 1023
380 POKE CHRAM + I, PEEK (CHROM + I)
390 NEXT I
400 REM
410 REM REDEFINO AL CONTROL B
420 REM
430 START = MEMTOP * 256 + (2 + 64) * 8
440 FOR I = 0 TO 7
450 READ A
460 POKE START + I, A
470 NEXT I
480 PRINT "MODIFICADO": ; CHR $(2)
490 END
500 REM
510 REM CONFIGURACION DEL NUEVO
520 REM CARACTER
530 REM
540 DATA 54, 108, 0, 118, 118, 126, 110, 0
```

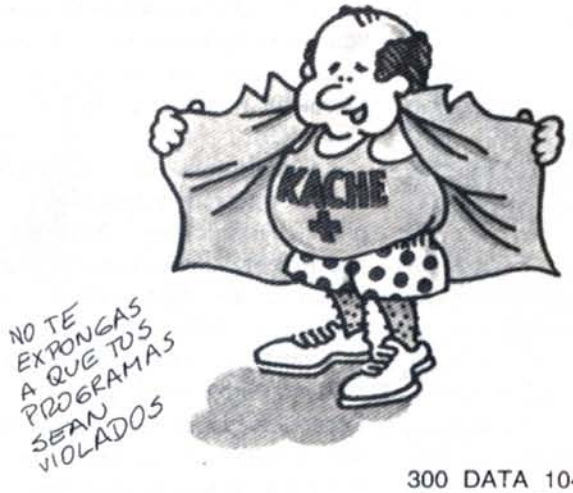
Desprotector y protector de listado

Un método de protección bastante común para los programas Basic es "trancar" el listado mediante el sistema que puedes encontrar en el número de línea 32767. Cuando se utiliza este sistema la única forma de ejecutar el programa es mediante las instrucciones RUN"C:" o RUN"D:NOMBRE", con lo que se logra evitar que alguien pueda ver o modificar el listado de dicho programa.

El método de desprotección que te estamos entregando te permite cargar el programa y luego mediante el uso de la tecla START "destrancar" el listado con la cual podrás listar y modificar dicho programa.

La única consideración que nos queda por ver es la posibilidad del "pirateo". Aunque esta sea una herramienta que permite listar un programa y eventualmente modificarlo, es ético respetar el trabajo de otros progamadores, ya que ellos viven de su actividad y lo que para ti puede ser un juego, para otros puede significar un desastre económico.

El programa que a continuación te entregamos, sirve para que puedas ver listados de programas, lo que sin duda es una de las mejores maneras de aprender a programar mejor.




```
100 REM
110 REM    REVISTA TURBO NEWS
120 REM
130 REM    DESPROTECTOR DE
LISTADOS
140 REM
150 RESTORE
160 POKE 710,0
170 FOR I=1 TO 65
180 READ B
190 POKE 255+I,B
200 NEXT I:X=USR(256)
210 DIM A$(20)
220 ? "ARCHIVO A CARGAR";
230 INPUT A$:? CHR$(125)
240 ? :? " AL APARECER LA
PALABRA 'READY'"
250 ? "PRESIONE LA TECLA START"
260 ? " LUEGO PUEDE LISTAR SU
PROGRAMA"
270 ? :? "...CARGANDO"
280 LOAD A$
290 END
300 DATA 104, 160, 11, 162, 1,
169, 7, 32, 92, 228, 96, 8, 72, 152,
72, 173, 31, 208, 201, 6, 208, 35,
169, 310 DATA 0, 168, 145, 138,
200, 169, 128, 145, 138, 200, 320
DATA 169, 6, 145, 138, 200, 145,
138, 200, 169, 33, 145, 138, 200,
169, 22, 145, 138, 173, 200, 2, 330
DATA 106, 234, 234, 234, 104, 168,
104, 40, 76, 98, 228, 0
```

Para proteger un archivo con el objeto de que no pueda ser listado, debes usar la rutina que se te entrega a continuación, la cual debes incorporar en tu propio programa. La forma correcta de su utilización es: GOTO 32767. Si deseas grabar tu programa en un disco, debes cambiar la instrucción SAVE"C:" por SAVE"D:NOMBRE".

```
32767 POKE PEEK (138) + PEEK
(139) * 256 + 2,0: SAVE "C:": NEW
```

ASSEMBLER ASSEMBLER

Por: Marcelo Waldbaum

LECCION 5

En esta sección del curso de Lenguaje Assembler analizaremos dos conceptos muy importantes dentro de la programación en Lenguaje de Máquina. En primer lugar, describiremos la utilización de la Pila y sus instrucciones PHA y PLA. Luego estudiaremos algunos de los distintos tipos de direccionamiento que nuestro procesador posee.

Dentro del Mapa de Memoria de nuestro computador Atari, en las direcciones comprendidas entre $ 100 y $ 200 (en decimal 256 a 511), encontramos lo que en Lenguaje de Máquina llamamos "La Pila".

Estas ubicaciones de memoria son utilizadas frecuentemente por el procesador cada vez que se ejecuta una instrucción JSR o bien RTS.

Supongamos que estamos en un Supermercado y vemos una pila de latas de café. La Pila de nuestro computador se asemeja a la pila de café. Si deseamos comprar una lata, debemos tomar la última que se encuentra en su extremo. Si los empleados del Supermercado, desean agregar más latas a la pila, deben hacerlo de a una y por encima de la que se encuentra en el extremo superior de la pila.

Cuando retiramos una lata de la pila, la antecúltima toma automáticamente la primera posición. Nunca se debe quitar una lata del medio o de abajo.

La Pila del procesador 6502 se comporta igual que la del Supermercado. Pero en vez de ser de latas, es una Pila de Bytes, que puede almacenar hasta un máximo de 256 Bytes o bien, 1 página como se suele decir en Assembler (1 página = 256 Bytes).

De las instrucciones que en Assembler pueden afectar el contenido de la Pila, estudiaremos en esta sección sólo cuatro de ellas: PHA, PLA, y reanalizaremos las instrucciones JSR y RTS.

INSTRUCCIONES PHA Y PLA:

Muchas veces es necesario guardar el contenido del acumulador o de algún registro para saltar a una subrutina y permitirle a ésta, modificar dicho registro, para luego restablecer su valor original. Esta tarea puede realizarse de diversas maneras. Si deseamos guardar sólo el acumulador podemos utilizar una posición de memoria. Un ejemplo de esta situación sería:

```
100 LDA #$A1
110 STA $2.000
120 JSR $E456
130 LDA $2.000
```

En la línea 100, cargamos el acumulador con el valor $A1, luego en la 110 almacenamos el acumulador en la dirección $2.000. Al llamar a la subrutina $E456, es muy probable que el

acumulador regrese de ésta con un valor modificado. Por lo tanto, en la línea 130 lo recuperamos. En este caso, la rutina $E456 es sólo un ejemplo de aplicación que en este caso no realiza ninguna función. El problema que presenta esta estructura radica en la cantidad de Bytes que estas instrucciones ocupan. La línea STA $2.000 ocupa 3 Bytes de programa, lo mismo que la línea 130.

Utilizando la instrucción PHA, guardamos el contenido del acumulador en la Pila, ocupando sólo 1 Byte de programa al realizarlo y con la instrucción PLA, recuperamos el mismo, también en un solo byte de instrucción. Otra de las ventajas que tiene este sistema, es la velocidad en la ejecución del programa. El procesador es considerablemente más rápido trabajando con la Pila con respecto a cualquier otra dirección de la memoria. El ejemplo anterior realizado con PHA y PLA tendría el siguiente desarrollo:

```
100 LDA #$A1
110 PHA
120 JSR $E456
130 PLA
```

Al almacenar más de un valor en la Pila es necesario tener mucho cuidado, pues de la Pila sale primero el último valor que entró en ella. Si por ejemplo, guardamos en la Pila los registros X e Y en este orden, al retirarlos debemos sacarlos en el orden Y - X:

```
100 LDX #$A1
110 TXA
120 PHA
130 LDY #$C9
140 TYA
150 PHA
160 JSR $E456
170 PLA
180 TAY
190 PLA
200 TAX
```

En este caso, utilizamos las instrucciones TXA, TYA, TAX y TAY, debido a que con la Pila sólo trabaja el acumulador y por lo tanto, no existe ninguna instrucción que guarde el contenido de los registros X e Y directamente en la Pila. Por eso hicimos el traspaso a través del registro A.

En la edición Nro. 2 de nuestra revista, explicamos la utilización de las instrucciones JSR y RTS. En este curso profundizaremos más este concepto, pues ambas instrucciones al ejecutarse afectan la estructura de la Pila.

Cuando una instrucción JSR es ejecutada por el procesador, éste en su Pila almacena la dirección de retorno. El 6502 se fija cuál es la dirección de la instrucción que sigue al JSR, le resta uno y almacena el Byte de bajo orden (LO-BYTE) de esta dirección en la Pila seguido del Byte de alto orden o HI-BYTE

ASSEMBLER ASSEMBLER

Al encontrar el RTS correspondiente en la subrutina que culmina, extrae el procesador de la Pila los Bytes de retorno para saber en qué dirección debe seguir ejecutando instrucciones.

Esta estructura de funcionamiento automático nos obliga a tener mucho recaudo con el trabajo que realicemos dentro de la Pila. Si se da el caso en que ejecutamos un llamado a subrutina y en ella ingresamos un valor en la Pila que luego no se extrae por error, al encontrar el RTS, el procesador obtiene como dirección de retorno el Byte que acabó de ingresar en la rutina, lo cual sería erróneo y generaría un salto a cualquier dirección dentro de la memoria generándose un error insalvable en el programa.

SISTEMAS DE DIRECCIONAMIENTO

Hasta esta sección del curso sólo hemos aplicado, aunque no lo habíamos mencionado, tres modos de direccionamiento: el Modo Implícito, el Absoluto y el Inmediato.

Llamamos Direccionamiento Implícito a aquellas instrucciones que normalmente afectan sólo a un registro o a un indicador del procesador.

Cuando ejecutamos la instrucción INX, en ella está implícito que afecta al registro X. No necesitamos ningún operando para aclararlo. Lo mismo sucede con la instrucción CLC que automáticamente afecta al indicador de acarreo.

Estas instrucciones ocupan siempre un solo Byte de programa, pues no necesitan operandos.

Las instrucciones que contienen Direccionamiento Absoluto, están compuestas por tres Bytes. El primero de ellos es el código de la instrucción y los dos restantes constituyen una dirección de memoria expresados en la forma LO-HI. Son ejemplos de Direccionamientos Absolutos:

```
LDA $A000
ADC $300
STY $400C
```

Las instrucciones que utilizan el modo de Direccionamiento Inmediato, son aquellas cuyo operando se encuentra definido en la misma instrucción y no se requiere un acceso de memoria para obtenerlo. En el caso de las instrucciones: LDX #$A2, LDA #$31 y ADC #$2, los operandos se definen por el byte, que sigue al signo "#". Con este tipo de direccionamiento sólo podemos trabajar con valores constantes, pues un LDA #$AF no considera la posición de memoria AF, sino realmente su valor hexadecimal.

En nuestro computador, la página cero tiene características especiales. Cuando utilizamos alguna posición ubicada en la página cero, es decir entre $00 y $FF, si convertimos estas posiciones en LO y HI Bytes, siempre el HI Byte es igual a cero. Lo podemos notar en la siguiente tabla:

DIRECCION	HI LO
$00	00 00
$01	00 01
$02	00 02
$FE	00 FE
$FF	00 FF

Como en este caso siempre estaríamos refiriéndonos a un valor 00 que ocuparía espacios ociosos, los diseñadores del 6502 definieron un tipo de direccionamiento de página cero que viene a ahorrar estos Bytes.

Este tipo de direccionamiento es muy similar al modo de Direccionamiento Absoluto, pero en lugar de 3 Bytes de programa ocupa sólo 2 Bytes. Cuando en Assembler realizamos LDA $80, el compilador analiza esta instrucción. Como detecta que es una dirección de página cero, en Lenguaje de Máquina genera el código A5 80.

El procesador no sólo utiliza menos Bytes para referirse a la página cero, sino que además, opera considerablemente más rápido con las instrucciones que usan el direccionamiento de página cero. Pero debido a esta situación, hay que tener cuidado con las direcciones de página cero que utilizamos, pues éstas generalmente son muy ocupadas por el Sistema Operativo del Atari y podrían traer serios inconvenientes en nuestros programas en Lenguaje de Máquina.

Dejamos para la siguiente sección del Curso, los sistemas de direccionamiento indexados, para luego ingresar de lleno en el desarrollo de programas completamente escritos en Assembler. Es la idea de TURBO news, terminar de explicar todos los conceptos teóricos de este lenguaje, para poder comenzar un desarrollo práctico de un programa de aplicación.

RANKING DEL MES

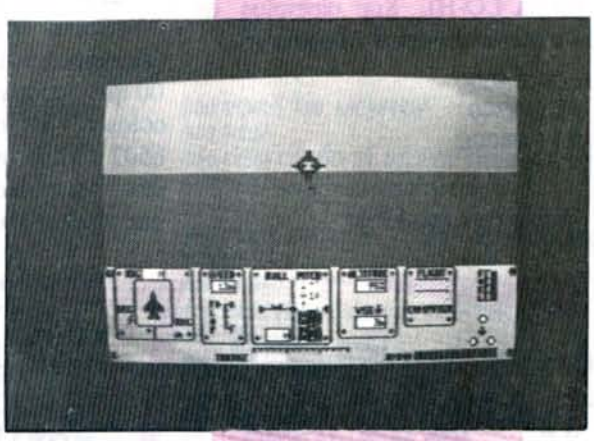
POS. DEL MES	POS. MES ANTERIOR	TITULO
1	15	STAR RAIDERS II
2	3	NINJA
3	12	LEADER BOARD GOLF
4	6	BMX SIMULATOR
5	24	FIGHTER PILOT II
6	21	FLAK
7	22	INTERNATIONAL KARATE
8	7	FUTBOL II
9		HENRY'S HOUSE
10	1	MONTEZUMA'S REVENGE
11	20	ELEKTRAGLIDE
12		DIMENSION X
13	13	GREAT AMERICAN RACE
14	14	NINJA MASTER
15		KIKSTART
16	18	SWAT
17	19	GHOSTBUSTERS
18		GHOSTCHASER
19		POOYAN
20	16	OIL'S WELLS
21	22	BOULDER DASH II
22	23	MARIO BROSS.
23		TANK COMMANDER
24		MIRAX FORCE
25	25	PACMAN

ASCENSO CONSTANTE DESCENSO

Continuando con la entrega de rankings mensuales, damos en este número el correspondiente al mes de septiembre, obtenido en base a estadísticas de venta de cassettes software en Chile. Recuerda que tus preferencias también serán tenidas en cuenta, para lo cual debes escribimos con los juegos de tu elección.

FIGHTER PILOT II

Este juego es otro de los buenos simuladores de vuelo. En este caso, piloteas un avión de caza cuya misión es defender el espacio participando en terribles combates aéreos. Este juego utiliza el joystick y el teclado conjuntamente. La potencia de las turbinas se maneja con las teclas Q para incrementarla y A para disminuirla. Para subir o bajar las ruedas se debe oprimir la tecla U. Para doblar a izquierda o derecha se puede usar el joystick o bien las teclas Z y X, logrando con estas últimas un viraje más suave. W y S manejan los flaps, utilizados en los aterrizajes para elevar la punta del avión. Con las teclas SELECT y OPTION debes encender la Computadora de Combate de tu nave. Con ella podrás averiguar la altura a la que se encuentra tu enemigo. Cuentas además en el tablero, con un indicador de la posición relativa de tu enemigo, la distancia a que se encuentra, indicador de daños, altura, combustible, brújula, velocidad, flaps, posición de las ruedas, municiones, horizonte, altura y demás instrumentos imprescindibles para la conducción de un avión caza.



SONIDO	8.5
GRAFICACION	8.7
JUGABILIDAD	8.8
PRESENTACION	8.7
PROMEDIO	8.8/5

LEADER BOARD GOLF

SONIDO	6.7
GRAFICACION	7
ADICCION	6.9
PRESENTACION	6.9
PROMEDIO	6.875

Una réplica exacta de lo que es un juego de Golf. Magníficos escenarios junto con intrincados hoyos hacen de éste, un apasionante juego. Podrás elegir, en cada tiro, entre cualquiera de los nueve hierros y tres maderas que el mismo posee, moviendo el joystick hacia arriba y abajo. Podrás jugar con o sin viento, dependiendo del nivel escogido, cuidando de no caer en los obstáculos naturales que cada una de las cuatro canchas posee. Para medir la fuerza de tus golpes deberás mantener oprimido el botón del joystick mientras un indicador de potencia que se encuentra a la derecha, te indicará el incremento de la misma. Cuando la misma alcanza el nivel requerido, debes soltar el botón ante lo cual el tiro se efectuará. Dicha fuerza más el palo escogido, harán que la pelota vaya a una mayor o menor distancia. Sin duda, uno de los simuladores mejor logrados para el computador ATARI.

STAR RAIDERS II

Este juego está basado en la película "El último Guerrero Espacial". Las fuerzas de KODAN intentan apoderarse de la Galaxia y nuestro deber es defenderla.

Afortunadamente, nosotros también conocemos la ubicación de su Galaxia, por lo que también podremos iniciar una ofensiva. Las batallas se desarrollan tanto en el espacio como en los planetas y los objetivos militares consisten en naves espaciales y bases terrestres. Para el combate, cuentas con un poderoso láser y radares de detección del enemigo. Luego de los combates, deberás controlar las averías de tu nave y, de ser necesario, dirigirte urgentemente a tu base para efectuar las correspondientes reparaciones.

SONIDO	6.4
GRAFICACION	6.9
ADICCION	6.7
PRESENTACION	6.7
PROMEDIO	6.675

APRENDIENDO A DIBUJAR CON LOGO

Ya hemos visto cómo dibujar con el LOGO y cómo se trabaja con el color del mismo. En este número, comenzaremos viendo cómo podemos modificar el dibujo de la Tortuga. Este lenguaje te permite tener hasta 16 figuras al mismo tiempo, una de las cuales es la tortuga que no se puede modificar. Sin embargo, las otras 15 se encuentran a tu disposición. El computador posee una cuadrícula de 16 filas por 8 columnas dentro de la cual podrás hacer tus propios dibujos.

0	1	2	3	4	5	6	7	8
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								

En la lección de hoy veremos cómo modificar el dibujo de la tortuga del logo por otro cualquiera que deseemos. Además empezaremos a usar construcciones de instrucciones que, sin duda, te serán de gran ayuda.

Por: Mauro Pieressa S.

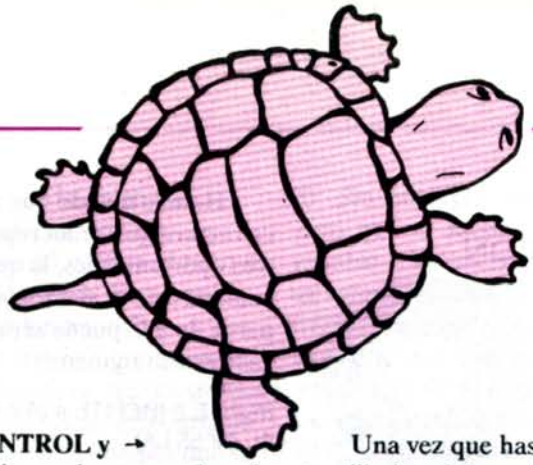
Cada uno de los cuadrados puede estar completamente pintado o bien completamente en blanco. No te preocupes si te parece que un dibujo va a quedar muy "cuadradote", ya que en la práctica, el dibujo va a ser mostrado en forma reducida y todas las cuadraturas desaparecerán. Para crear o modificar una figura se utiliza la instrucción EFOR.

INSTRUCCION EFOR nro.

Su significado es "Editar forma número", donde número es un valor que va de 1 a 15 e indica cuál de las 15 grillas para gráficas disponibles deseamos crear o modificar. Ejemplo:

EFOR 1

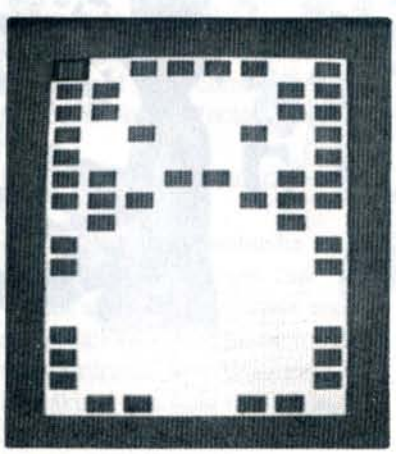
Luego de ejecutada esta instrucción, todo lo que hay en pantalla desaparecerá y en su lugar se verá una cuadrícula sobre la cual podrás hacer tus dibujos. En la esquina superior izquierda, te aparecerá el cursor. Para pasar a otra casilla deberás oprimir las siguientes teclas:



Tecla CONTROL y →
Para desplazar el cursor a derecha
Tecla CONTROL y ←
Para desplazar el cursor a izquierda
Tecla CONTROL y ↑
Para desplazar el cursor hacia arriba
Tecla CONTROL y ↓
Para desplazar el cursor hacia abajo

Para cambiar la casilla que está bajo el cursor, de vacío a ocupado y viceversa, deberás oprimir la Barra Espaciadora. Si deseas salir del modo de edición de figuras, puedes oprimir la tecla ESC para indicar que te interesa que se almacene la figura creada. Si quieres salir sin guardar la figura deberás oprimir la tecla BREAK.

A continuación, te damos un ejemplo de un dibujo que podrás incorporar a los tuyos:



Una vez que has concluido tus dibujos, debes avisarle al computador qué forma tomarán cada una de las tortugas. Debes tener en cuenta que si no le das ninguna instrucción, el computador le asignará a las cuatro tortugas la figura 0 que es justamente la tortuga. Para darle ese aviso, se utiliza la instrucción PONFR.

INSTRUCCION PONFR nro.

Su significado es "Pon forma". Luego de ejecutada esta instrucción se le asignará a la o las tortugas activas, la forma correspondiente al número nro. Ejemplo:

```
PONFR 2
```

La o las tortugas activas tomarán la forma de la figura 2.

Hasta ahora habrás notado lo molesto que es dibujar un cuadrado, por ejemplo, ya que debes repetir cuatro veces lo mismo. Ejemplo:

```
LM
AV 100
DE 90
AV 100
DE 90
AV 100
DE 90
AV 100
DE 90
```

Existe, sin embargo, una instrucción que permite evitar este tipo de repeticiones. La misma es REPITE.

INSTRUCCION REPITE nro [instrucciones]

Su significado es "repite un número de veces las instrucciones que se encuentran entre corchetes". Por ejemplo, las instrucciones del último ejemplo para dibujar un cuadrado, pueden reemplazarse por las siguientes:

```
LM
REPITE 4 [AV 100 DE 90]
```

Finalmente, en este número te enseñaremos la instrucción PONXY. Muchas veces habrás querido empezar un dibujo en

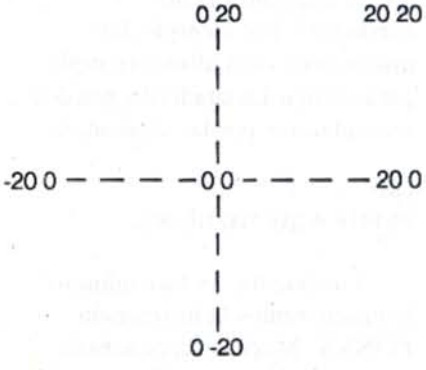
APRENDIENDO A DIBUJAR CON

LOGO

algún lugar específico y te habrás demorado varias instrucciones en poner allí la tortuga. Para ello, existe esta instrucción.

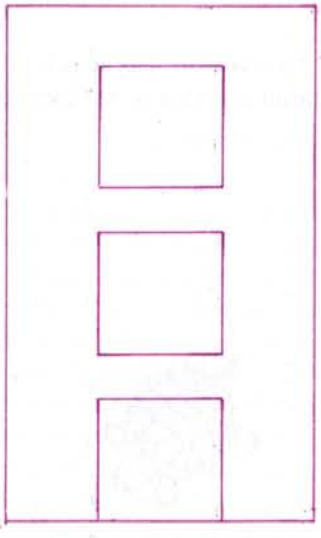
INSTRUCCION PONXY [nro1 nro2]

Su significado es "pon la tortuga en la posición número 1 y número 2". El número 1 se refiere a la posición horizontal y el número 2 a la vertical. Para entenderlo mejor es que te presentamos el siguiente gráfico. Debes tener en cuenta que cuando comienza el programa o cuando ejecutamos la instrucción LM la tortuga se posiciona en el lugar [0 0].



A continuación te damos un ejemplo de aplicación para las instrucciones dadas en este número.

```
LM
SL
PONXY [-50 -60]
LA
REPITE 2 [AV 170 DE 90 AV 100 DE
90]
DE 90
AV 30
IZ 90
REPITE 4 [AV 40 DE 90]   (1)
SL
AV 55
LA
REPITE 4 [AV 40 DE 90]
SL
AV 55
LA
REPITE 4 [AV 40 DE 90]
```



Habrás notado que hay una estructura que se ha repetido en tres oportunidades, la que comienza en el símbolo (1). A partir de allí, puede ser reemplazada por lo siguiente:

```
REPITE 3 [REPITE 4 [AV 40 DE 90]
SL AV 55 LA]
```

Se va a ejecutar 1 vez de más lo siguiente:

```
SL
AV 55
LA
```

Pero no importa porque no va a ocasionar ningún cambio al dibujo final.

Con el uso, podrás comprobar la gran ayuda que te dará la instrucción REPITE. En el próximo número continuaremos en el aprendizaje de este entretenido lenguaje.

TRUCOS CON EL JOYSTICK

DESARROLLANDO HARDWARE

Por: Pedro P. Caraball A.

Pues a mí me ha pasado y luego de algunos experimentos logré desarrollar divertidas variaciones que requieren poco dinero y proveen lindos resultados.

1.- BASTA DE DISCRIMINACION CONTRA LOS ZURDOS!!!

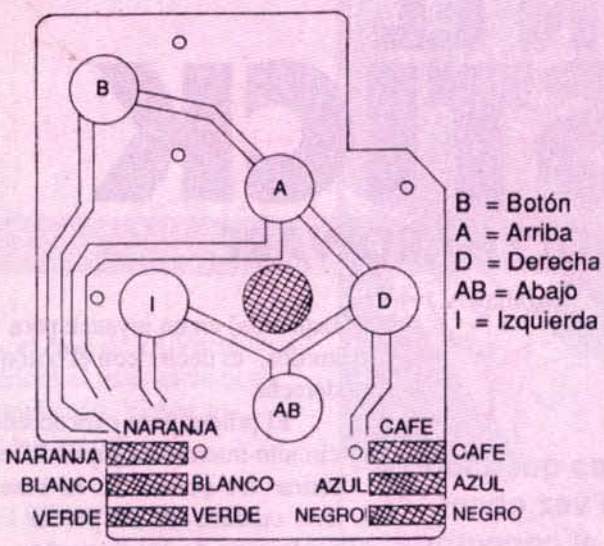
Como viejo y zurdo fanático de los juegos, desde hace tiempo sospechaba que mis scores eran perjudicados por el hecho de que los joysticks fueron diseñados para diestros.

Cavilando sobre el tema, descubrí que es bastante simple convertir un joystick Atari Standard (con disparador a la izquierda) en un joystick para zurdos, es decir, con el botón a la derecha.

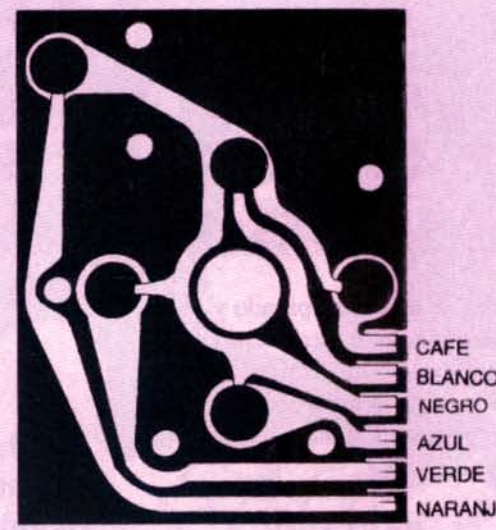
¿Te has quedado alguna vez observando el conector para joysticks de tu computador, pensando cuántas clases de hardware exótico podrían conectarse a él?

El primer paso para acceder al circuito interno del joystick, es retirar los pernos de la base. Ten cuidado de no perder cualquiera de los pequeños componentes que rodean el botón disparador. Coloca el circuito de manera tal que se asemeje a la figura 1.

Joystick Standard



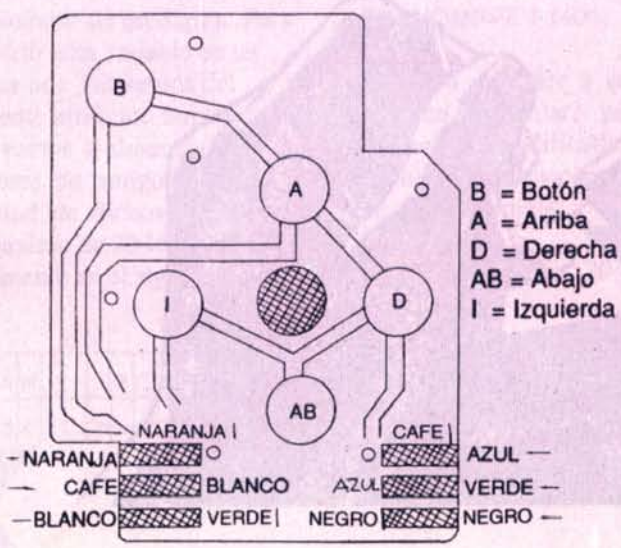
Joystick Nuevo



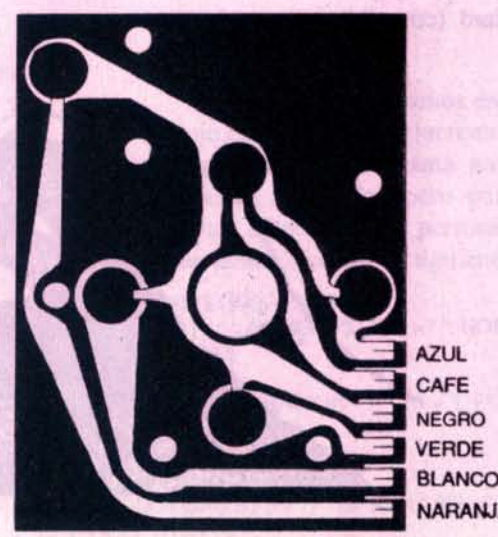
Nota: Los joystick Atari más modernos, tienen todos los conectores en un lado de la plaqueta, mientras que los más antiguos poseen tres conectores de cada lado.

Los conectores del cable deben ser quitados de la plaqueta con delicadeza, sin tirar de los cables. Luego, debes armar el circuito según las conexiones indicadas en la figura 2.

Joystick standard para zurdos



Joystick nuevo para zurdos



Como verás, lo único que hay que hacer, es reubicar los conectores en sus nuevas posiciones dependiendo del color de los cables.

La única precaución que te sugerimos, es etiquetar debidamente tu nuevo joystick de "zurdos", para evitar enloquecer a un pobre diestro inadvertido.

2.- UN DISPARADOR CASERO POR POCOS PESOS:

Si acaso tienes un joystick que conoció mejores días pero que está listo para pasar a retiro, puedes reencarnarlo convirtiéndolo en un "joystick a botonera".

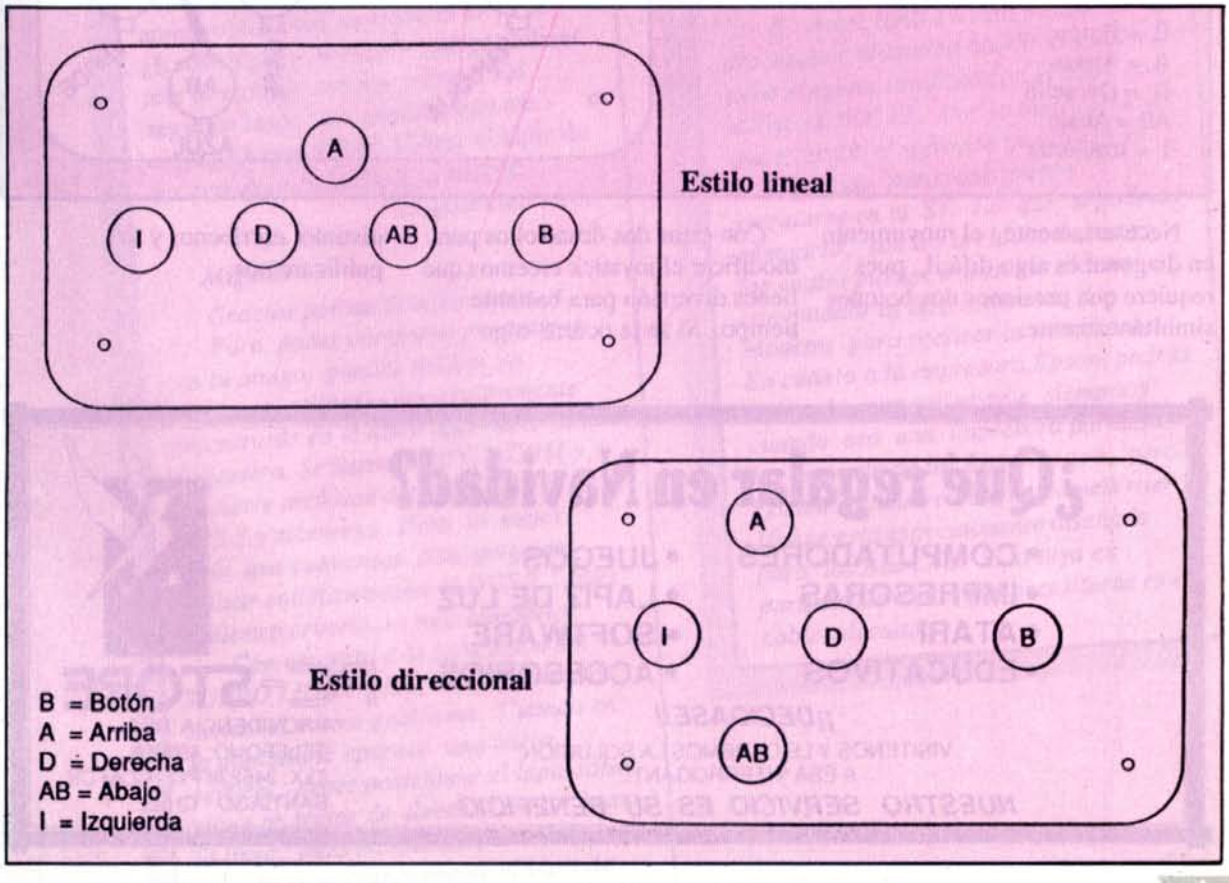
La idea es simple: se trata de crear un panel de botones para controlar todas las funciones del joystick. Seguramente habrás notado que en los locales de juegos electrónicos la mayoría de las máquinas carecen de palancas de control puesto que funcionan en base a este tipo de paneles.

Para llevar a cabo este desarrollo sólo requieres unos pocos componentes disponibles en el mercado:

Un cable de joystick.
Algunos interruptores o botones.
Cables cortos,
y una caja de plástico para montar el desarrollo.

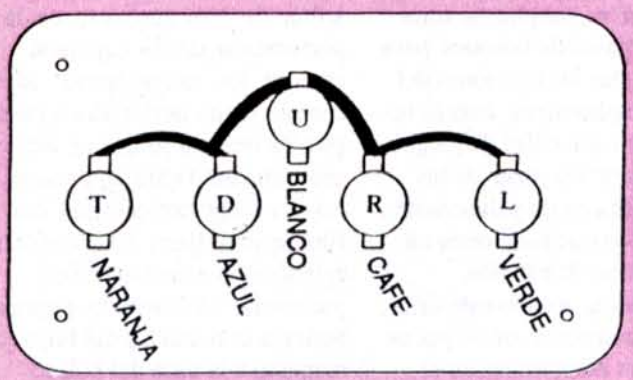
Probablemente la parte más ardua de este proyecto es la perforación de la caja para insertar los interruptores. Si careces de un perforador a mecha, puedes intentar lo que yo hice: usar un cautil para empezar el hoyo y luego completarlo con el filo de una tijera. El secreto de esta técnica es trabajar con paciencia, chequeando permanentemente el diámetro del hoyo con respecto a la base del botón.

En las siguientes figuras podrás elegir la configuración que prefieras para tu proyecto.

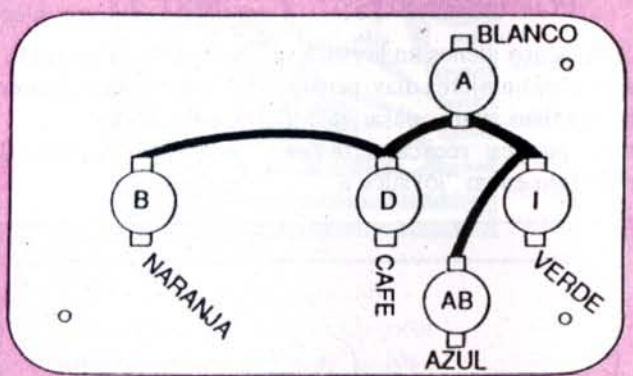


Debes también perforar la caja en algún costado con el objeto de permitir la salida del cable conector.

Para conectar los cables del joystick a cada uno de los botones deberás seguir el siguiente esquema:



Circuito de conexionado lineal



Circuito de conexionado direccional

B = Botón
A = Arriba
D = Derecha
AB = Abajo
I = Izquierda

Necesariamente, el movimiento en diagonal es algo difícil, pues requiere que presiones dos botones simultáneamente.

Con estos dos desarrollos para modificar el joystick creemos que tienes diversión para bastante tiempo. Si se te ocurre algo distinto, escríbenos y lo publicaremos.

¿Qué regalar en Navidad?

- COMPUTADORES
- IMPRESORAS
- ATARI
- EDUCATIVOS
- JUEGOS
- LAPIZ DE LUZ
- SOFTWARE
- ACCESORIOS

¡¡DECIDASE!!

VISITENOS Y LE DAREMOS LA SOLUCION A ESA INTERROGANTE

NUESTRO SERVICIO ES SU BENEFICIO

PC-STORE

PROVIDENCIA 1449
TELEFONO: 499908
TLX: 346236 PC - COM CK
SANTIAGO - CHILE

ATARI sounds

Por: Marcelo Waldbaum

Una de las herramientas más importantes en el desarrollo de programas para el Computador Atari, es la utilización del potencial que éste posee en cuanto a la generación de sonidos.

Para esto, el Computador Atari posee un sistema de hardware especializado, que se encuentra centralizado en un Integrado llamado POKEY. Para generar sonidos con el computador, no es necesaria la incorporación de ningún amplificador o periférico a nuestro equipo. Lo único que se requiere es entender el funcionamiento de la instrucción SOUND del ATARI BASIC.

El sistema de sonido de nuestro computador tiene la ventaja de poseer cuatro canales para producir sonidos. Esto nos da la posibilidad de generar acordes de cuatro notas distintas, o bien utilizar cada uno de los canales para generar un tipo de sonido que se utilice en determinada situación dentro del programa.

Cada canal es independiente del otro y en cada uno de ellos, podremos controlar su volumen, su frecuencia y su nivel de distorsión. La instrucción Basic que activa cada uno de los cuatro canales tiene la siguiente forma:

```
10 SOUND P1,P2,P3,P4
```

Los parámetros P1,P2,P3,P4 deben ser números enteros. P1 identifica el canal que va a ser utilizado y sus valores posibles son: 0,1,2 y 3. P2 representa la frecuencia del sonido que se quiere obtener. Sus valores posibles oscilan entre 0 y 255. El parámetro P3 define qué tipo de sonido estamos utilizando. Ocho tipos de sonido podemos definir con este parámetro, dos de los cuales definen notas puras y los restantes obtienen sonidos distorsionados. Con estos, podemos obtener ruidos interesantes, tales como motores,

ATARI sounds

sonidos de helicópteros, disparos de ametralladoras, etc. P4 define el volumen del sonido emitido por el canal en curso. Dieciséis volúmenes distintos pueden obtenerse con los valores que van desde el 0 (nulo) al 15 que representa el volumen máximo.

Es importante recalcar que la suma de los volúmenes de los cuatro canales no debe superar el valor de 32, pues se generaría una sobremodulación en la salida del sonido por el parlante del televisor.

Como podemos ver, el número de sonidos generables con el Computador Atari, sólo está limitado por la imaginación del programador, pues podemos variar notas, volúmenes, superponer canales con notas distintas e inclusive, tenemos la posibilidad de utilizar canales con ruidos extraños a partir del valor de distorsión definido en la instrucción SOUND.

En este artículo, vamos a incorporar una serie de Subrutinas de sonidos especiales que podrás agregar en tus programas para así obtener un mejor producto de software. Lo único que debes hacer es ingresar estas líneas en tus listados.

RUIDO DE UNA ALARMA:

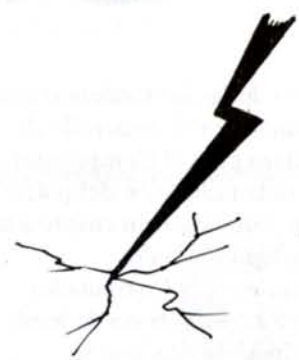
```
100 DUR=15
110 FOR T=1 TO DUR*12
120 V=15
130 I=0.5
140 L=3
150 GOSUB 220
160 NEXT T
170 V=10
180 I=0.95
190 L=1
200 GOSUB 220
210 END
220 SOUND 0,53,10,V
230 SOUND 1,60,10,V
240 V=V*I
250 IF V>L THEN 220
260 RETURN
```

SIRENA POLICIAL:

```
100 DUR=15
110 LO=50
120 HI=35
130 ST=-1
140 FOR T=1 TO DUR
150 FOR N=LO TO HI STEP ST
160 SOUND 0,N,10,14
170 FOR I=1 TO 20
180 NEXT I
190 NEXT N
200 X=LO
210 LO=HI
220 HI=X
230 ST=-ST
240 NEXT T
250 END
```

SONIDOS DE RELAMPAGOS:

```
100 ST=RND(0)*5+0.2
110 FOR P=5 TO 100 STEP ST
120 VOL=(RND(0)*10+5)/(0.1*P)
130 SOUND 0,P,8,VOL
140 SOUND 1,P+20,8,VOL
150 NEXT P
160 GOTO 100
```



EXPLOSION:

```
100 DUR=15
110 N=20
120 GOSUB 140
130 END
140 SOUND 2,75,8,15
150 I=0.79+DUR/100
160 V1=15
170 V2=15
180 V3=15
190 SOUND 0,N,8,V1
200 SOUND 1,N+20,8,V2
210 SOUND 2,N+50,8,V2
220 V1=V1*I
230 V2=V2*(I+0.05)
240 V3=V3*(I+0.08)
250 IF V3>1 THEN 190
260 SOUND 0,0,0,0
270 RETURN
```

TIMBRE DE UNA PUERTA:

```
100 FREC=105
110 DUR=7.5
120 GOSUB 170
130 FREC=132
140 DUR=8.5
150 GOSUB 170
160 END
170 VOL=15
180 ICR=0.79+DUR/50
190 SOUND 0,FREC,10,VOL
200 VOL=VOL*ICR
210 IF VOL>1 THEN 190
220 RETURN
```

VUELO DE UNA MOSCA:

```
100 P=INT(RND(0)*6)+250
110 V=INT(RND(0)*4)+6
120 SOUND 0,P,14,V
130 FOR X=1 TO 10
140 NEXT X
150 GOTO 100
```

Promoción Extraordinaria
En este precio ganas:

Sólo por $5.940.- valor equivalente a 12 números de tu revista "TURBO news", ahora recibes 13 ediciones.

En el cupón de suscripción indícanos desde qué número deseas suscribirte y recibirás el ejemplar extra que tú desees.

- Descuento de un 10% del valor de tu revista TURBO news.
- 1 número extra (13 ediciones por el valor de 12).
- Mantención del precio durante el período de suscripción.
- Gastos de despacho certificado incluido.

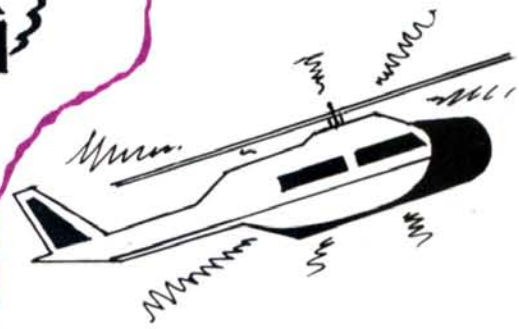
Llena el cupón de suscripción ahora y envíalo a: Editora Turbo Ltda. Av. Holanda 2456 correo 9 Santiago, adjuntándonos cheque cruzado y nominativo. Indícanos a nombre de quién facturamos.

SONIDO DEL TELEFONO:

```
100 FOR I=1 TO 35
110 R=0.3
120 L=2
130 GOSUB 250
140 NEXT I
150 R=0.9
160 L=1
170 GOSUB 250
180 SOUND 0,0,0,0
190 SOUND 1,0,0,0
200 T=T-1
210 IF T<1 THEN 100
220 FOR J=1 TO 300
230 NEXT J
240 GOTO 100
250 VOL=15
260 SOUND 0,40,10,VOL
270 SOUND 1,42,10,VOL
280 VOL=VOL*R
290 IF VOL>L THEN 260
300 RETURN
```

RUIDO DE HELICOPTERO:

```
100 FOR A=1 TO 10
110 SOUND 0,A,A,A
120 NEXT A
130 GOTO 100
```



Es la intención de TURBO news que nos ayudes a formar una biblioteca de programas con sonidos. Así, cuando necesitemos uno determinado, lo ubicaremos rápidamente en nuestra Revista. Para esto, te solicitamos que nos escribas a la Editora Turbo Limitada, Holanda 2456, Providencia, Santiago, con tus rutinas de sonidos, las cuales formarán parte de futuras publicaciones.

La nueva generación de Software para Computadores Atari

ADQUIERALOS EN LOS SIGUIENTES PUNTOS DE VENTAS

• **ANTOFAGASTA:** COOPERCARAB / KW VIDEO / LA ESPAÑOLA • **VIÑA DEL MAR:** FALABELLA VIÑA / INSIS / MPR COMPUTACION • **VALPARAISO:** COMPUTRONIC • **SANTIAGO:** AUDIO BICICLETA INTERNAC / CASA ROYAL / CENTRO ATARI / COMERCIAL ESTADO / COMPUMANQUE / COMPUCENTER / FALABELLA AHUMADA / FALABELLA P. ARAUCO / IMACO / INFOGROUP / PC STORE / PETERSEN / ROLEC / SUPERMERCADOS UNIMARC / TASCO / VIDEO CLUB INTERNACIONAL • **RANCAGUA:** CASA ZUÑIGA • **CURICO:** MULTIHOGAR • **TALCA:** LIBRERIA "EL AHORRO" / MULTICENTRO / VIDEO CLUB CASSAL • **CHILLAN:** CASA EDISON • **CONCEPCION:** COOPERCARAB / DISMAR / DISMAR 2 / EQUS / PHANTER / RAPSODIA / SESCO • **LOS ANGELES:** DISTRIBUIDORA MERINO • **ANGOL:** SCORPIO • **VICTORIA:** CASA SIGMUND • **TEMUCO:** COMERCIAL MANQUEHUE / ESTABLECIMIENTOS GEJMAN / FALABELLA / • **PUCON:** ELTIT • **VILLARRICA:** JOYERIA KETTERER • **VALDIVIA:** ELECTROMUSICA • **LA UNION:** IMPORTADORA COSMOS • **OSORNO:** CASA REAL / FOTO EXPRESS • **PUERTO VARAS:** ELECTRO HORN • **PUERTO MONTT:** COMERCIAL MANQUEHUE / DIMARSA • **COYHAIQUE:** FACI HOGAR • **PUNTA ARENAS:** BALFER LTDA.

JUEGOS

PARA COMPUTADORES ATARI

ADQUIERELOS EN LOS SIGUIENTES PUNTOS DE VENTAS

• **ANTOFAGASTA:** COOPERCARAB KW VIDEO LA ESPANOLA • **VINA DEL MAR:** FALABELLA VINA INSIS MPR COMPUTACION • **VALPARAISO:** COMPUTRONIC • **SANTIAGO:** AUDIO BICICLETA INTERNAC CASA ROYAL CENTRO ATARI COMERCIAL ESTADO COMPUMANQUE COMPUCENTER FALABELLA AHUMADA FALABELLA P ARAUCO IMACO INFOGROUP PC STORE PETERSEN ROLEC SUPERMERCADOS UNIMARC TASCO VIDEO CLUB INTERNACIONAL • **RANCAGUA:** CASA ZUNIGA • **CURICO:** MULTIHOGAR • **TALCA:** LIBRERIA "EL AHORRO" MULTICENTRO VIDEO CLUB CASSAL • **CHILLAN:** CASA EDISON • **CONCEPCION:** COOPERCARAB DISMAR DISMAR 2 EQUS PHANTER RAPSODIA SESCO • **LOS ANGELES:** DISTRIBUIDORA MERINO • **ANGOL:** SCORPIO • **VICTORIA:** CASA SIGMUND • **TEMUCO:** COMERCIAL MANQUEHUE ESTABLECIMIENTOS GEJMAN FALABELLA • **PUCON:** ELITE • **VILLARRICA:** JOYERIA KETTERER • **VALDIVIA:** ELECTROMUSICA • **LA UNION:** IMPORTADORA COSMOS • **OSORNO:** CASA REAL FOTO EXPRESS • **PUERTO VARAS:** ELECTRO HORN • **PUERTO MONTT:** COMERCIAL MANQUEHUE DIMARSA • **COYHAIQUE:** FACI HOGAR • **PUNTA ARENAS:** BALFER LTDA